

Tuning em Banco de Dados

Data base Tuning

Ana Paula dos Santos Souza¹

Bruno Fidelis Campos¹

Carla Glênia Guedes Dias¹

Michel Batista Alves¹

Carlos Eduardo Costa Vieira²

Flávio Campos Carelli³

Luiz Fabiano Costa de Sá³

Artigo
Original

Original
Paper

Palavras-chaves:

Otimização

Consultas

T-SQL

Microsoft SQL Server
2005

Resumo

Devido ao grande volume de dados que são gerados pelas Empresas que utilizam Sistemas de Informação, é fundamental o papel do Banco de Dados (BD). Geralmente os dados precisam ser acessados a todo instante, logo, a disponibilidade dos resultados nem sempre são satisfatórias. Nesse contexto, entra a questão do desempenho ao se obter informações de um BD e como otimizá-las. Muitos problemas de performance não estão relacionados a infraestrutura, sistemas operacionais ou mesmo ao hardware. Pode-se encontrar problemas de perda de performance dentro do próprio BD, sendo a consulta a principal causadora desses problemas. Ajustar e otimizar uma consulta e o próprio BD tornam-se fatores importantes, podendo-se ter um ganho de performance aceitável, visto que cada consulta é tratada de forma diferente, dependendo do Sistema Gerenciador de Banco de Dados (SGBD). Este artigo avalia como melhorar o desempenho de consultas Transact-Structured Query Language (T-SQL) em um ambiente Microsoft SQL Server 2005, sugerindo possíveis alterações que possam levar a um ganho de performance considerável.

Abstract

Due to large volume of data generated by companies that use information systems, the role of Database (DB) is fundamental. In general, data must be accessed at any time and the availability of results are not always satisfactory. In this context, begins the question of the performance in obtain information from a DB and optimize them. Many performance problems are not related to the infrastructure, operating systems or even the hardware. You can encounter problems of performance within DB, and queries are primary cause of these problems. Adjust and optimize queries and the DB becomes an important factor, it may have a acceptable gain of performance, since each query is treated differently depending on the Data Base Management System (DBMS). This paper evaluates how to improve performance of Transact-Structured Query Language (T-SQL) in a Microsoft SQL Server 2005, suggesting possible changes that could lead to a considerable gain in performance.

Key words:

Optimization

Queries

T-SQL

Microsoft SQL Server
2005.

¹ Discente do Curso de Sistemas de Informação – UniFOA

² Doutor e Docente do Curso de Sistemas de Informação – UniFOA

³ Docente Especialista do Curso de Sistemas de Informação – UniFOA

1. Introdução

O mercado atual está competitivo. Com isso, as empresas estão apostando, cada vez mais, em sistemas informatizados que forneçam apoio à melhoria de seus processos.

Dentro desse contexto, surgem os Sistemas de Gerenciamento de Banco de Dados (SGBD's) com o intuito de armazenar e gerenciar as informações, garantindo sua disponibilidade de forma rápida e eficaz. Mas, em geral, SGBD's não são ferramentas auto suficientes no que diz respeito a otimização de consultas a Banco de Dados (BD). Por esse motivo, esforços são despendidos na forma de aperfeiçoar seu funcionamento interno, melhorando a organização das informações e como são obtidas.

Segundo Ikematu (2003), tuning é a sintonia ou ajuste de algo para que funcione melhor. O tuning fornece suporte ao Administrador de BD (Database Administrator ou DBA) através de um mecanismo que simplifica a análise de desempenho, fazendo com que pequenos ajustes afetem significativamente a performance do BD, transformando uma tarefa de alto custo e complexidade em um processo simples e rápido.

Neste artigo serão demonstrados procedimentos de como se pode realizar a identificação de onde está o problema de performance da aplicação, otimizações de consultas Transact-Structured Query Language (T-SQL) e boas práticas ao elaborá-las.

Para uma análise sobre o desempenho das consultas SQL em SGBD's, foi escolhido o Microsoft SQL Server 2005, "pois é muito utilizado pelas empresas além de possuir ferramentas que facilitam uma auditoria das consultas que estão sendo realizadas" (ANDRADE, 2005, p. 10).

Este artigo está organizado da seguinte maneira: A Seção 2 apresentará conceitos e definições de tuning. A Seção 3 descreverá a definição dos problemas de desempenho de um BD. A Seção 4 mostrará como identificar problemas de performance no SQL Server. A Seção 5 apresentará algumas ferramentas de apoio na identificação de gargalos no SQL Server. A Seção 6 descreverá a otimização de consultas T-SQL. A Seção 7 apresentará boas práticas na elaboração de consultas. Por último, a Seção 8 mostrará as considerações finais e possíveis propostas de trabalhos futuros.

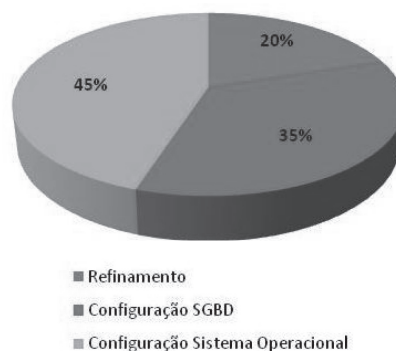
2. Conceitos e Definições de Tuning

Segundo Baptista (2008, p. 15), "tuning diz respeito ao ajuste do SGBD para melhor utilização dos recursos, provendo um uso eficaz e eficiente do SGBD".

A fase de tuning de um BD é um processo de refinamento que envolve modificações em vários aspectos, abordando desde mudanças nos conceitos aprendidos nos Diagramas Entidade-Relacionamento (DER) até a troca de hardware, passando pela configuração dos softwares que executam nesse sistema.

Em termos didáticos, pode-se dividir as ações de tuning em três grandes tipos: (1) refinamento do esquema das relações e as consultas/atualizações feitas no BD, (2) configuração do sistema operacional em uso e (3) configuração dos parâmetros dos SGBD's (TRAMONTINA, 2008, p. 2). O foco do artigo será no primeiro tipo.

Em debate com profissionais com mais de 10 anos de experiência em gerenciamento de banco de dados, os Srs. Lúcio Gomes Peixoto Júnior (Microsoft Certified Trainers, Microsoft Certified Systems Engineer, Microsoft Certified Technical Specialist, Microsoft Certified IT Professional e Oracle Certified Professional) e Erick de Souza Carvalho (Mestre em Engenharia da Computação pelo Instituto Tecnológico da Aeronáutica e Oracle Certified Professional), constata-se que 45% das ações de tuning são destinadas a configuração do Sistema Operacional, 35% destinadas a configuração do SGBD e 20% destinadas ao refinamento das consultas, como mostra a Figura 1.



Fonte: Peixoto Júnior e Carvalho (2008)
Figura 1 – Fases do Tuning

3. Problema

Antes de pesquisar os problemas de desempenho nas consultas SQL, é vital saber se realmente há problemas. Se existem, é preciso identificar corretamente a causa. Uma vez verificado que existem problemas nas consultas à base de dados, é preciso saná-los, e, realmente, as consultas SQL são responsáveis por boa parte dos problemas de desempenho das aplicações que utilizam BD (ANDRADE, 2005).

Atualmente o desempenho de um BD é um fator que determina efetivamente sua disponibilidade. O objetivo do tuning é fornecer suporte ao DBA através de um mecanismo que simplifique a análise de desempenho em BD, transformando uma tarefa de alto custo e complexidade em um processo simples e rápido (DIAS, 2005, p. 1).

3.1 Problemas de Desempenho

Segundo Andrade (2005), o principal problema relacionado aos BD's é que eles são ambientes dinâmicos. Também, a população de dados cresce constantemente e suas configurações podem mudar.

Dessa forma, complica-se a análise ou previsão do desempenho. Geralmente os problemas de desempenho de um BD estão relacionados a índices e consultas SQL, como mostra a Tabela 1.

PROBLEMAS DE DESEMPENHO
70% a 80% de todos os problemas de desempenho em aplicações que utilizam BD são causados por consultas SQL mal feitas (MULLINS, 1998, apud ANDRADE, 2005, p. 13)
Instruções SQL e índices são responsáveis por 60% a 90% dos problemas de desempenho de aplicações (LECCO, 2003, apud ANDRADE, 2005, p. 13);
Atividades relacionadas às consultas SQL consomem entre 70% e 90% dos recursos dos BDs (LECCO, 2003, apud ANDRADE, 2005, p. 14);

Tabela 1 – Problemas de Desempenho

Apesar dos cuidados com o desempenho do BD iniciarem desde a sua concepção, passando pelo design e normalização das tabelas

até a sua configuração, são as consultas que determinam quão bom é o desempenho do BD.

“Dentre os problemas de desempenho que um SGBD abrange, estão incluídos o consumo de processamento, utilização ineficaz de comandos SQL, bloqueios, esperas e atividade de disco” (DIAS, 2005, p. 1).

4. Identificando Gargalos no Microsoft SQL Server

Segundo Biggs e Venezia (2006, p. 1), “gargalos significam restrições em determinadas formas de comunicação, interação ou transferência de informações.”

O principal problema dos gargalos de desempenho nas empresas (em particular na área de Tecnologia da Informação) é que eles podem ser difíceis de identificar. Alguns são mais óbvios do que outros.

Segundo Fernandes (2006), para encontrar possíveis gargalos e processos com problemas no SQL Server, é recomendado que se utilize a tabela de sistema sysprocesses, como mostra a Tabela 2.

PASSOS
<pre>select * from master..sysprocesses where status = 'runnable' ou select * from master..sysprocesses where dbid = xx</pre> <p>(Este passo serve quando há suspeita de um gargalo em um BD)</p>
<p>Para saber o DataBase ID (DBID) de um BD, deverá ser executado o seguinte comando:</p> <pre>select DB_ID('nome do BD)</pre> <p>ou então deve-se executar diretamente a query passando o nome do BD como parâmetro:</p> <pre>select * from master..sysprocesses where dbid = db_id('Pubs')</pre> <p>Esta query retornará o status de cada processo no SQL Server: spid kpid blocked waittype waittime lastwaittype waitresource dbid uid ...</p> <p>Convém observar se existe algum tempo em que um processo está em espera, verificando o status waittime, waittype ou lastwaittype. Identificando o tipo de espera, é possível saber o que está acontecendo, se o problema é de Entrada/Saída (E/S), lock de um objeto (LATCH), ou memória (MEMORY), etc.</p>

PASSOS

É recomendado utilizar o comando Database Consistence Check (DBCC) e SQLPERF (threads) por último. Este comando fornece o status de cada thread aberta pelo SQL Server, bem como cada processo associado a esta thread, retornando as seguintes informações de status: E/S, CPU (processador) e MEMORY.

Fonte: Fernandes (2006)

Tabela 2 – Passos para Encontrar Possíveis Gargalos

Sysprocesses é uma tabela do sistema que contém informações sobre o Server Process ID (Servidor de Identificação de Processos ou SPID) ativo, que estão em execução no SQL Server (MICROSOFT, 2007).

Existem outros pontos de gargalo que são decorrentes de ajustes no sistema, que consomem mais tempo de processamento (processador), E/S e memória, cujas possíveis causas são:

- Alteração ou atualização que possa ter levado a uma queda no desempenho do sistema, ou seja, em relação ao software: alguma nova instalação ou atualização. Já em relação ao banco de dados: adição e/ou remoção de bases de dados, funções, procedimentos, consultas, tabelas, linhas, colunas e etc. É de extrema importância o planejamento de mudança e sua documentação, seja ela no software ou no BD;
- Limitação de hardware, ou seja, trabalhando próximo de sua capacidade máxima;
- Consultas mal elaboradas.

5. Ferramentas de Apoio

Existem ferramentas disponibilizadas pelo SGBD que fornecem melhorias de desempenho nos sistemas de bancos de dados já existentes. A seguir serão apresentadas breves descrições das ferramentas que estão presentes no Microsoft SQL Server 2005.

5.1 SQL Server Profiler

Segundo Moraes (2007, p. 4), o SQL Server Profiler (Figura 2):

é utilizado para capturar as informações que estão chegando no banco de dados. Trata-se de uma ferramenta extremamente útil para detecção de problemas de performance, depurar a aplicação ou mesmo entender de que maneira e quando as aplicações estão em interação com o SQL Server, sem a necessidade de abrir o código fonte da aplicação.

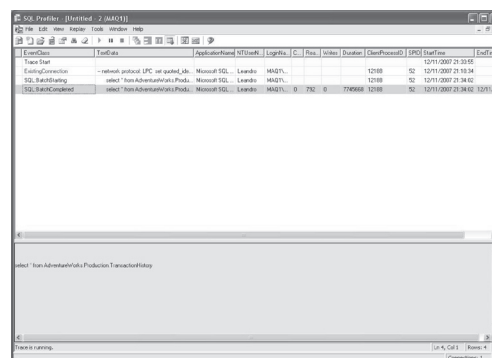


Figura 2 – SQL Server Profiler

5.2 Database Engine Tuning Advisor

Segundo Moraes (2007, p. 18), o Database Engine Tuning Advisor (DTA) (Figura 3):

é uma poderosa ferramenta que auxilia os Administradores de Banco de Dados na seleção apropriada de projeto físico de uma instalação do Microsoft SQL Server 2005. O DTA pode ser utilizado tanto para ajuste de pequenas consultas que possam estar mal desenhadas e que apresentem problemas de desempenho, quanto para consultas mais complexas e que requerem maior conhecimento do DBA em relação ao desenho físico do banco de dados.

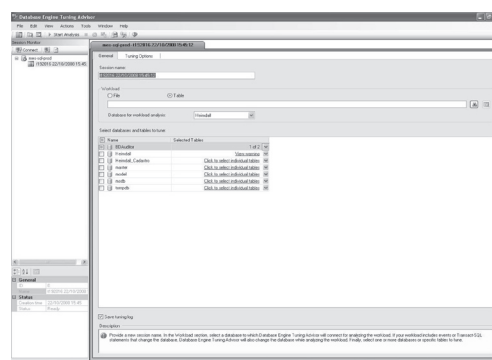


Figura 3 – Database Engine Tuning Advisor

5.3 Execution Plan

Segundo Moraes (2007, p. 10), o Execution Plan (Figura 4):

provê um diagrama visual de como o SQL Server está executando uma determinada consulta, quais índices estão sendo utilizados, além de uma série de outras informações que podem auxiliar os DBA's no processo de otimização de índices ou outros dados requeridos pelo SQL Server para aumentar o desempenho das consultas.

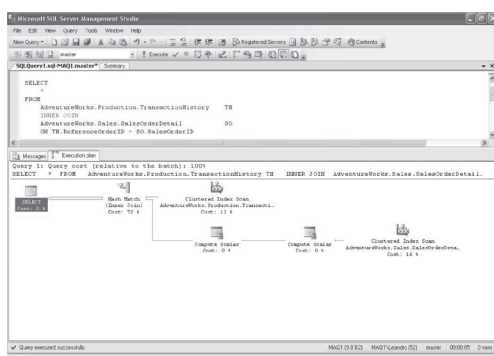
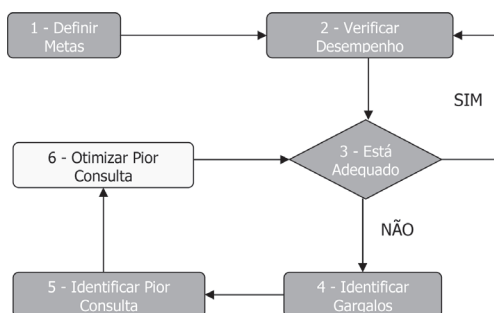


Figura 4 – Execution Plan

6. Otimização de Consultas T-SQL

A Figura 5 apresenta um fluxograma ilustrando a otimização de consultas T-SQL.

Segundo os passos mostrados no fluxograma, pode-se obter uma redução de custos e melhoria de qualidade, que está relacionada à disponibilidade da informação em tempo hábil, permitindo tomadas de decisões mais rápidas (SILVA, 2006, p. 6).



Fonte: Silva (2006)

Figura 5 – Fluxograma para Otimização de Consultas

7. Boas Práticas na Elaboração de Consultas T-SQL

A Tabela 3 sugere boas práticas ao se trabalhar com consultas T-SQL no SQL Server 2005 com o objetivo de melhorar significativamente o desempenho do BD.

BOAS PRÁTICAS EM CONSULTAS T-SQL

Procurar escrever consultas levando em consideração boas práticas de desenvolvimento, tais como indentação e comentários (/* */ ou --).

Informar os campos que devem aparecer na consulta, evitando utilizar o famoso select * from <NomeDaTabela>.

Quando for utilizar operadores de comparação, evitar usar NOT em condições de pesquisa, pois podem diminuir a velocidade de recuperação de dados porque todos os registros em uma tabela são avaliados.

Usar de forma restritiva a cláusula WHERE, pois é uma grande causadora de problemas em relação ao processador.

Para testes de existência é sempre mais eficiente utilizar EXISTS ao invés de COUNT. Quando se utiliza COUNT, o BD não sabe que está sendo feito um teste de existência e continua pesquisando todas as linhas qualificadas. Já utilizando EXISTS, o BD sabe que é um teste de existência e interrompe a pesquisa quando encontra a primeira linha qualificada.

Analisar a necessidade de utilização de CHAR e VARCHAR.

Analisar a possibilidade de se usar o operador UNION ALL em substituição ao UNION – DISTINCT.

As Procedures sempre devem retornar um valor de status de execução.

Utilizar COUNT(1) ou COUNT (NomeDoCampo) ao invés de COUNT(*) .

Utilizar as funções COALESCE e ISNULL.

Não chamar funções SQL repetidamente, ao em vez disso, deve-se armazenar o valor em uma variável.

Utilizar variáveis do tipo TABLE em substituição a tabelas temporárias – Recompilações.

Usar SET NOCOUNT em Procedures, Functions e Triggers.

BOAS PRÁTICAS EM CONSULTAS T-SQL
Utilizar índices sempre que necessário.
Em relação aos parâmetros, o ideal é garantir que tenha o mesmo datatype da coluna com a qual ele será comparado.

Tabela 3 – Boas Práticas na Elaboração de Consultas

8. Considerações Finais

Para a elaboração deste artigo foi realizado um estudo teórico sobre tuning em Banco de Dados, demonstrando sua importância para o desenvolvimento e manutenção de aplicações. É necessário uma monitoração constante assim como, ajuste e administração do ambiente, consistindo de políticas, procedimentos, ferramentas e utilitários integrados de gerenciamento de performance.

Para conseguir melhores resultados no acesso ao Banco de Dados, critérios de desempenho devem ser considerados nas fases iniciais do desenvolvimento de software. Com isso também evita-se que seja necessário um grande esforço com re-projeto ou re-codificação para se atingir um nível de desempenho satisfatório e nem sempre consegue-se os mesmos resultados se os critérios de performance fossem considerados desde o início do desenvolvimento.

A utilização de boas práticas em Banco de Dados traz benefícios para a organização, pois o tempo de resposta em uma consulta T-SQL é minimizado e a performance é garantida.

Como trabalhos futuros, poderia-se utilizar o SGBD Oracle ou qualquer outro SGBD para avaliação em termos de tuning.

9. Referências

ANDRADE, L. D. Otimização de Consultas de Aplicações T-SQL em Ambiente SQL Server 2000. 2005. 52 f. Monografia (Graduação em Ciência da Computação) – Departamento de Ciência da Computação, Instituto de Matemática, Universidade Federal da Bahia, Salvador, 2005. Disponível em: <http://disciplinas.dcc.ufba.br/pub/MATA67/TrabalhosSemestre20051/Monografia_Luciano_Andrade.pdf>. Acesso em: 05 maio 2008.

BAPTISTA, C. de S. Administração de Sistemas de Gestão de Banco de Dados. 2008. Disponível em: <www.dsc.ufcg.edu.br/~baptista/cursos/ABD/ADM1.ppt>. Acesso em: 07 maio 2008.

BIGGS, M.; VENEZIA, P. Combata os Gargalos de Desempenho. 2006. Disponível em: <http://computerworld.uol.com.br/gestao/2006/04/17/idgnoticia.2006-04-17.9538042938/IDGNoticia_view>. Acesso em: 07 maio 2008.

DIAS, E. S. DB Guardian: Sistema para Diagnóstico de Desempenho em Banco de Dados. Escola Regional de Banco de Dados, UFRGS, 2005. Disponível em: <<http://www.inf.ufgrs.br/erbd/Artigos/poster/7489.pdf>>. Acesso em: 04 maio 2008.

FERNANDES, R. Identificando Gargalos no SQL Server 2000 – I. 2006. Disponível em <<http://www.mcdbabrazil.com.br/modules.php?name=News&file=article&sid=358>>. Acesso em: 07 maio 2008.

IKEMATU, R. S. Realizando Tuning na Base de Aplicações. 2003. Disponível em: <http://www.devmedia.com.br/articles/viewcomp_forprint.asp?comp=11323>. Acesso em: 07 maio 2008.

LECCO Technology. Performance Tuning for Mission-Critical Database Applications. 2003. Disponível em: <<http://www.questsoftware.com.br/documents/landing.aspx?id=8947&technology=47&prod=&prodfamily=&loc=>>>. Acesso em: 01 ago. 2008.

MICROSOFT. Description of the waittype and lastwaittype columns in the master.dbo.sysprocesses table in SQL Server 2000 and SQL Server 2005. 2007. Disponível em: <<http://support.microsoft.com/kb/822101/en-us/>>. Acesso em: 07 maio 2008.

MORAES, A. J. de J. Análise das Ferramentas Database Engine Tuning Advisor, SQL Server Profiler e Execution Plain na Melhoria de Desempenho do Microsoft SQL Server 2005. 2007. Disponível em <http://iremar.prof.sites.uol.com.br/Publicacoes/PosGraduacao2007/Altair_TCC.pdf>. Acesso em: 11 nov. 2008.

MULLINS, C. S. SQL Analysis and Review. 1998. Disponível em: <http://www.craigs-mullins.com/sql_a-r.htm>. Acesso em: 05 maio 2005.

PEIXOTO JÚNIOR, L. G.; CARVALHO, E. de S. Informação Verbal. 2008.

TRAMONTINA, G. B. Database Tuning: Configurando o Interbase e o PostgreSQL. Campinas, 2008. Disponível em: <<http://www.ic.unicamp.br/~geovane/mo410-091/Ch20-ConfigInterbasePosgres-art.pdf>>. Acesso em: 29 abr. 2008.

SILVA, C. B. Otimizando o Desempenho de um Banco de Dados em um Ambiente Altamente Crítico. 2006. 79f. Monografia (Bacharel em Sistemas de Informação) - Departamento de Sistemas de Informação, UNIMINAS, Uberlândia, 2006. Disponível em: <<http://si.uniminas.br/TFC/monografias/MONOGRAFIA-CLESIO.pdf>>. Acesso em: 30 abr. 2008.

Endereço para Correspondência:

Carlos Eduardo Costa Vieira
Sistemas de Informação – UniFOA
cadu.vieira@gmail.com

Centro Universitário de Volta Redonda
Campus Três Poços
Av. Paulo Erlei Alves Abrantes, nº 1325,
Três Poços - Volta Redonda / RJ
CEP: 27240-560

Informações bibliográficas:

Conforme a NBR 6023:2002 da Associação Brasileira de Normas Técnicas (ABNT), este texto científico publicado em periódico eletrônico deve ser citado da seguinte forma: SOUZA, Ana Paula dos Santos; CAMPOS, Bruno Fidelis; DIAS, Carla Glênia Guedes; ALVES, Michel Batista; VIEIRA, Carlos Eduardo Costa, CARELLI, Flávio Campos. E LUIZ FABIANO COSTA DE SÁ. Tuning em Banco de Dados. **Cadernos UniFOA**. Volta Redonda, ano IV, n. 10, agosto. 2009. Disponível em: <http://www.unifoa.edu.br/portal_pesq/caderno/edicao/10/19.pdf>